

# 11. Tushar

**Program Name:** Tushar.java

**Input File:** tushar.dat

Tushar really enjoys working with 2-D arrays and is always looking for new ways to work with the array data. Traditional uses typically involve horizontal and vertical processing but Tushar is trying a diagonal pattern and is having some difficulty. Starting at the top left corner, which is indexed as row 0 and column 0, work down and right to compute the average of the elements along a diagonal. Then, work across the top to generate an average for each down and right diagonal. Continue by working down the left side and generate an average, again moving down and right along each diagonal.

Col → ↓ Row	0	1	2	3	4	5
0	1	6	11	15	18	20
1	21	2	7	12	16	19
2	25	22	3	8	13	17
3	28	26	23	4	9	14
4	30	29	27	24	5	10

In the above example with 5 rows and 6 columns, the first shaded diagonal contains 1...5, the second diagonal 6...10, the third shaded diagonal 11...14, the fourth diagonal 15...17, the fifth shaded diagonal 18...19, and finishing across the top with just 20. Working down the left side would be 21...24, then shaded 25...27, then 28...29, and finally just the shaded 30. This produces 10 diagonals with averages shown in sample output below.

**Input:** First line contains a single integer **N** the number of test cases that follow with  $N \leq 10$ . Each test case starts with a line containing 2 integers separated by whitespace: **R**, the number of rows, and **C**, the number of columns, with both  $2 \leq R, C \leq 12$ . That line will be followed by **R** lines of data with each containing **C** integers separated by whitespace containing integers in  $[-100, 100]$ .

**Output:** For each test case, output 1 row of averages, separated by single spaces. Display the averages with 2 digits after the decimal point.

**Sample input:**

```

3
5 6
1 6 11 15 18 20
21 2 7 12 16 19
25 22 3 8 13 17
28 26 23 4 9 14
30 29 27 24 5 10
4 4
-13 -9 -5 -1
-14 -10 -6 -2
-15 -11 -7 -3
-16 -12 -8 -4
3 2
1 -6
-2 5
3 -4
    
```

**Sample output:**

```

3.00 8.00 12.50 16.00 18.50 20.00 22.50 26.00 28.50 30.00
-8.50 -6.00 -3.50 -1.00 -11.00 -13.50 -16.00
3.00 -6.00 -3.00 3.00
    
```